Admission Credit Model – Probability of Default

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Developed for: | CEO, CFO, Risk Control, Accountants |
| Level: | Confidential |
| Document Owner: | Chief Financial Officer / Chief Credit Officer |
| Created: | Januari 2024 |
| Regulation: | IFRS9 – Accounting standard.  EBA GL 2017 06 – EBA guidelines on implementing IFRS9 in credit institutions |

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Creator | Date | Description |
| 2.0 | Data Scientist | Januari 2024 | Document Creation |

Table of Contents

[1. Definitions & Shortages 5](#_Toc157679556)

[2. Introduction and Background 6](#_Toc157679557)

[3.Algorithms 7](#_Toc157679558)

[3.1.1 Logistic Regression 7](#_Toc157679559)

[3.1.2 Decision Tree 8](#_Toc157679560)

[3.1.3 Random Forrest 8](#_Toc157679561)

[3.1.4 Gradient Boosting 9](#_Toc157679562)

[3.2 Hyperparameter Tunning 10](#_Toc157679563)

[4. Evaluation Metrics 10](#_Toc157679564)

[4.1 Evolution Metrics – Hyperparameter Tunning 11](#_Toc157679565)

[Admission Scorecard 12](#_Toc157679566)

[5. Data 12](#_Toc157679567)

[5.1 Sources 12](#_Toc157679568)

[5.2 Data Selection: Co - Applicant 12](#_Toc157679569)

[5.3 Target Feature 13](#_Toc157679570)

[5.4 Time Selection – data splits 13](#_Toc157679571)

[5.5 Balancing 13](#_Toc157679572)

[6. Feature Selection 14](#_Toc157679573)

[Top Features - Histograms 16](#_Toc157679574)

[7. Model Selection 17](#_Toc157679575)

[8. Results 18](#_Toc157679576)

[8. Selection of model continued – variables 19](#_Toc157679577)

[9. Calibration 20](#_Toc157679578)

[10. Risk Grouping 22](#_Toc157679579)

[Behavior Scorecard 24](#_Toc157679580)

[11. Data – BSC 24](#_Toc157679581)

[11.1 Source 24](#_Toc157679582)

[11.2 Target Feature 24](#_Toc157679583)

[11.3 Feature Importance 24](#_Toc157679584)

[12. Model Selection – BSC 25](#_Toc157679585)

[13. Results – BSC 27](#_Toc157679586)

[14. Calibration – BSC 28](#_Toc157679587)

[15. Features - BHC 29](#_Toc157679588)

[16. Staging & Significant Increase in Credit Risk 30](#_Toc157679589)

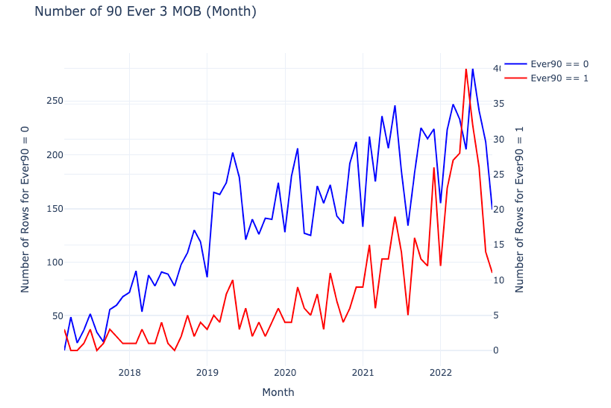
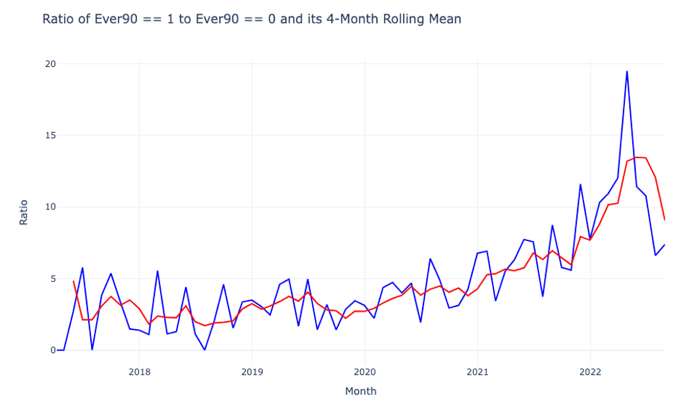
# Definitions & Shortages

|  |  |
| --- | --- |
| Term | Description |
| **IFRS 9** | Regulations containing principles for the classification and valuation of financial assets and liabilities, as well as provisions and write-downs of credit losses. |
| ECL – Expected credit loss | Expected credit loss (ECL) is an estimate of potential losses on loans and financial assets due to borrower defaults or payment shortfalls, used for accounting and risk assessment. |
| EAD - Exposure at default | Exposure at Default (EAD) is the amount of exposure or financial risk associated with a borrower or counterparty at the moment of default. |
| PV – Present Value | Present value monthly is the calculation of the current value of future cash flows, factoring in monthly interest rates. It helps determine the worth of future money in today's terms. |
| CR – Cure Rate | The "cure rate" refers to the percentage of defaulted loans or assets that return to a performing or non-default status, resulting in successful recoveries. It essentially measures the effectiveness of the efforts to bring defaulted loans or assets back to a healthy and performing state. |
| LGD - Loss Given Default | Loss Given Default (LGD) quantifies the expected financial loss in the event of a borrower's default on a loan, expressed as a percentage of the outstanding debt that cannot be recovered. |

# Introduction and Background

The previous scorecards at Nstart were developed for IFRS9 legislation reservation and not optimized for the purpose of qualitative credit scoring. Due to significant increase in credit losses relative to early years of the company decision has been taken to develop new scorecards developed after more recent data.

In this development different combinations of sampling, scaling, feature, hyperparameter tunning and algorithms has been tested. The results show significant improvements in both the admission & behaviors score cards.



*Plot showing show 12 month losses has increased over time*

# 3.Algorithms

Non-linearity, with its ability to capture intricate relationships and patterns in data that linear models cannot, underscores the importance of experimenting with various machine learning models. Embracing this diversity ensures that we can effectively tackle the complex and dynamic nature of real-world data, leading to improved model performance, robustness, and problem-specific solutions. Therefore, a number of different models will be applied and not only the traditional within banking which is to apply Logistic Regression.

# 3.1.1 Logistic Regression

Nature:

LG (Logistic Regression) is a supervised learning algorithm primarily employed for binary classification tasks.

Model Function:

Its core function is to utilize a logistic or sigmoid function to predict the probability that a given instance belongs to a specific class. The probability (P(Y=1|X)) of a particular class label being 1, given the input features (X), is calculated using the logistic function. This function depends on model parameters (β0, β1, …, βn) which are adjusted iteratively during the training process.

Objective:

The primary objective of LG is to optimize these parameters to maximize the Log-likelihood across all observations, essentially making the observed outcomes most probable according to the model. The Log-Likelihood serves as a measure of the model's predictive accuracy, and the optimization process aims to maximize it.

Estimation:

Parameter estimation typically employs Maximum Likelihood Estimation (MLE), adjusting the model parameters to make the observed outcomes most probable based on the model's predictions.

Output:

In practice, LG provides a probability score for each observation, allowing for classification by applying a threshold (commonly 0.5) to categorize instances into one of the two classes.

Assumption:

An underlying assumption of LG is that there exists a linear relationship between the log-odds of the outcome and the predictor variables, making it a powerful tool for a wide range of classification tasks.

# 3.1.2 Decision Tree

Nature:

Decision Trees (DT) are versatile supervised learning algorithms used for both regression and classification tasks.

Model Function:

DTs are represented as tree-like structures where each node represents a feature, each branch represents a decision rule, and each leaf represents an outcome. Decision-making in DTs involves splitting data based on feature values and selecting thresholds to maximize information gain (classification) or minimize impurity (classification and regression).

Advantages:

Transparency: DTs are easy to understand and interpret, making them suitable for explaining model predictions.

Minimal Data Preprocessing: They require little data preprocessing, handling both numerical and categorical data with ease.

Drawbacks:

Overfitting: DTs can overfit, particularly on small datasets or with deep trees, leading to poor generalization.

Bias with Imbalanced Data: DTs can be biased when dealing with imbalanced datasets, favoring the majority class.

Structural Variability: Small changes in input data can result in significant structural variations in DTs, reducing their robustness.

# 3.1.3 Random Forrest

Nature:

Random Forest (RF) is an ensemble learning algorithm that combines multiple decision trees to address classification and regression tasks.

Model Function:

RF aggregates predictions from a collection of decision trees. Each tree is trained on a different random subset of the data and features, and predictions are made by averaging (for regression) or voting (for classification) across the individual trees.

Advantages:

Improved Generalization: RF reduces overfitting compared to single decision trees, leading to improved generalization.

Robust to Noise: RF can handle noisy data and outliers due to the ensemble averaging effect.

Feature Importance: RF provides feature importance scores, aiding in feature selection and interpretation.

Drawbacks:

Complexity: RF can be computationally intensive, especially with a large number of trees.

Less Interpretability: While single decision trees are interpretable, RF's ensemble nature makes it less straightforward to explain.

Model Size: RF models can be large, which may impact memory usage and deployment.

In summary, Random Forest is a powerful ensemble learning method that mitigates the drawbacks of individual decision trees while providing robust predictions for a variety of tasks. It offers improved generalization, robustness to noise, and feature importance assessment, making it a valuable addition to your machine learning toolbox.

# 3.1.4 Gradient Boosting

Nature:

Gradient Boosting (GB) is an ensemble learning technique that combines multiple weak learners, often decision trees, to create a powerful model for both regression and classification tasks.

Model Function:

GB builds an ensemble of decision trees sequentially. It starts with an initial weak learner and then fits subsequent trees to the errors of the preceding ones. Predictions are made by aggregating the individual tree predictions, typically weighted by their contribution to the model.

Advantages:

High Accuracy: GB often achieves high predictive accuracy, as it iteratively corrects errors made by previous models.

Versatility: It can handle both regression and classification tasks effectively.

Feature Importance: GB provides feature importance scores, aiding in feature selection and interpretation.

Drawbacks:

Complexity: GB models can be computationally expensive and require careful tuning of hyperparameters.

Overfitting: If not properly regularized, GB can overfit, especially with deep trees.

Interpretability: While GB can provide feature importance, it may be less interpretable than simpler models like linear regression.

In summary, Gradient Boosting is a robust ensemble learning technique that excels in predictive accuracy and can handle various types of tasks. It leverages the strength of multiple weak learners to create a powerful model while offering insights into feature importance for enhanced understanding of the data.

# 3.2 Hyperparameter Tunning

Hyperparameter tuning is the process of systematically searching and optimizing the hyperparameters of a machine learning model. Hyperparameters are parameters that are not learned from the data but are set prior to the model's training. They include settings such as learning rates, regularization strengths, and the depth of decision trees.

Hyperparameter tuning is crucial because it helps improve the model's performance and generalization. Here's why it helps:

Optimizing Model Performance: Different hyperparameter values can significantly impact a model's performance. Tuning these hyperparameters allows us to find the configuration that leads to the best possible results on a given dataset.

Avoiding Overfitting: Hyperparameters like regularization strength can control a model's complexity. Tuning them appropriately helps prevent overfitting, where a model fits the training data too closely and performs poorly on unseen data.

Enhancing Generalization: A well-tuned model generalizes better to new, unseen data. Hyperparameter tuning ensures that the model is robust and capable of making accurate predictions on real-world data beyond the training set.

Efficient Resource Usage: Tuning hyperparameters can also improve the efficiency of model training. By finding the right set of hyperparameters, we can reduce the time and computational resources needed for training while still achieving excellent results.

In summary, hyperparameter tuning is a crucial step in the machine learning pipeline because it enables us to optimize model performance, enhance generalization, and efficiently allocate resources, ultimately leading to better and more reliable models.

# Evaluation Metrics

A confusion matrix is a fundamental tool in evaluating the performance of a machine learning model, especially in classification tasks. It provides a clear and concise summary of how well the model is classifying instances.

At its core, a confusion matrix breaks down the model's predictions into four categories:

* True Positives (TP): Instances correctly predicted as positive.
* True Negatives (TN): Instances correctly predicted as negative.
* False Positives (FP): Instances incorrectly predicted as positive (Type I error).
* False Negatives (FN): Instances incorrectly predicted as negative (Type II error).

From this matrix, several crucial metrics can be derived, each offering unique insights into the model's performance:

Accuracy: Measures the overall correctness of the model's predictions, calculated as (TP + TN) / (TP + TN + FP + FN).

Precision: Indicates the proportion of true positive predictions among all positive predictions, calculated as TP / (TP + FP). It measures the model's ability to avoid false positive errors.

Recall (Sensitivity or True Positive Rate): Represents the proportion of true positive predictions among all actual positive instances, calculated as TP / (TP + FN). It assesses the model's ability to identify all relevant positive cases.

F1-Score: Combines precision and recall into a single metric, providing a balanced measure of a model's performance. It is calculated as 2 \* (Precision \* Recall) / (Precision + Recall).

Specificity (True Negative Rate): Indicates the proportion of true negative predictions among all actual negative instances, calculated as TN / (TN + FP). It assesses the model's ability to identify all relevant negative cases.

False Positive Rate (FPR): Measures the proportion of false positive predictions among all actual negative instances, calculated as FP / (TN + FP). It is the complement of specificity.

## Evolution Metrics – Hyperparameter Tunning

Given the significant imbalance in the input data for the Validation/Test samples, assessing metrics in isolation, such as Accuracy and Recall, might initially seem satisfactory. Nevertheless, opting for metrics that strike a balance in evaluating the performance of both classes, like the F1-Score, can yield a more robust and dependable model assessment. To achieve this a hyperparameter class have been developed to boost Precision, F1 score while still aim for a satisfactory Accuracy.

For the final model evaluation, a comprehensive set of metrics has been employed to ensure a thorough assessment of its performance. These metrics include:

AUROC (Area Under the Receiver Operating Characteristic Curve): AUROC is a widely used metric that measures the model's ability to distinguish between positive and negative classes. It plots the true positive rate against the false positive rate, providing a comprehensive view of the model's discriminatory power.

PR Curve (Precision-Recall Curve): The PR curve focuses on the precision and recall trade-off, particularly valuable when dealing with imbalanced datasets. It offers insights into how well the model identifies positive instances while minimizing false positives.

KS Statistic (Kolmogorov-Smirnov Statistic): KS statistic quantifies the maximum difference between the cumulative distributions of positive and negative instances generated by the model. It helps identify the optimal threshold for classification and assesses the model's discriminatory power.

GINI Index: GINI is another metric used to evaluate the performance of classification models. It measures the inequality in the predicted probabilities of the two classes. A higher GINI index indicates better separation between classes.

By applying these metrics, we aim to provide a comprehensive and well-rounded evaluation of the final model's performance, ensuring that it not only classifies accurately but also excels in terms of discrimination and precision-recall trade-offs. This approach ensures that the model's strengths and limitations are thoroughly assessed, contributing to a more informed decision-making process.

# Admission Scorecard

# Data

## 5.1 Sources

The sources of data are the Credit Report from UC containing public data of the applicants that contain features like TimeOnAddress, NumberOffInquieries or the super variable UC Blanco Score (New supervariabels since Februari 2024). Apart from Credit Report additional data point from the core banking system Vilja has been used. These are populated from the customer when applying for the loan.

PSD2 data has not been utilized in this model due to that there is not same amount of history as the traditional credit data as mentioned abouve.

## 5.2 Data Selection: Co - Applicant

Since CO-applicants are a large part of the portfolio, they have been carefully handled in the way to adjust for selection of their data point. It has been tested to only use the main applicant’s data, only the Co- applicants UC score and the rest data points from the main applicant and finally all datapoints from the applicant with the best UC -score. The last experiment showed the highest dependency and will be applied. It was also tested to take both scores and put in to an additional model. This did not surpass the performance from just taking the score of the best applicant.

En bild som visar linje, Graf, diagram, sluttning

Automatiskt genererad beskrivning

*Main/CO applicant Delinquency*

## 5.3 Target Feature

The target feature will be if the applicant went delinquent (90+) after 12 months. It will be a binary classification where a positive label will be 1 as delinquent and 0 if healthy. The input features will be all other applied in the training of the models.

## 5.4 Time Selection – data splits

In the reporting database there is data from summer 2017 until today. Different constellations have been tested with different years included in the development data. From 2019 and onwards showed highest dependency and will therefore be applied.

The history of Nstart is quite scares in data and heavily imbalanced for the positive class over time. In other words, it can be relatively many positives in some periods while almost none in others. Since it’s important that the model is trained on both classes so it can balance the representation of all features the training, validation and Test datasets has been split after quartiles in the data to reassure balance.

Initially the data is sorted on date, then based only on the position of the positive quartile (Qt) the data is cut in less than 70 Qt for train, 70-85 for validation and 85 – 100 for test. The development data was from January 1th 2019-01 – 2022-11.

## 5.5 Balancing

Before training the models, the training data has been up sampled. This means that the positive data rows have been copied to balance the positive / the negatives. In other words, the minority class has been up sampled. The reason for this is that if not the weight will not be relative to the features of both classes it often leads to poorer performance of the algorithms applied.

# Feature Selection

To find strong features all potential has initially been included where a normalized feature importance of each algorithm has been stacked in order to see how the different algorithms handles the data and how the features over all perform. The normalization is done by minmax Scaling that crunches the values between 0 and 1 to make the feature importance between the algorithms comparable.

En bild som visar skärmbild, linje, Graf, diagram

Automatiskt genererad beskrivning

After this step only the top 50 was continued with and a correlation matrix performed on the rest. If assets were higher than 0.85 in correlation they were clustered together and the feature with the highest cumulative feature importance was kept. E.g. Income is a feature that comes in many variations, they all capture same thing in one way or another and will therefore be highly correlated.

En bild som visar text, skärmbild, diagram, Färggrann

Automatiskt genererad beskrivning

*Correlation Plot (Reduced #Features)*

## Top Features - Histograms

By plotting histograms of features with highest feature importance (defaulted vs not) one se strong visual behavior. Inquiries 12 Month, Age and the super variable UC Score are the most dependent features.

En bild som visar Graf, linje, diagram, text

Automatiskt genererad beskrivning

En bild som visar skärmbild, Graf, diagram, linje

Automatiskt genererad beskrivning

En bild som visar skärmbild, linje, Graf, diagram

Automatiskt genererad beskrivning

En bild som visar Graf, diagram, linje, skärmbild

Automatiskt genererad beskrivning

En bild som visar Graf, linje, skärmbild, diagram

Automatiskt genererad beskrivning

# Model Selection

After Hyper tunning to aim for optimized F1- score these where the results. Logistic Regression showed to show highest dependency.

En bild som visar skärmbild, diagram, linje, Färggrann

Automatiskt genererad beskrivning

In terms of AUROC & PR (Classification) Curves Logistic Regression showed the most stable results.

En bild som visar text, diagram, linje, Graf

Automatiskt genererad beskrivning

# Results

For the final evaluation the selected model from model evaluation showed better performance than the UC Blanco model in Train and Test sample by 2 points both in AUC & GINI. The Validation set was 1 point below the Blanco model, it should however be mentioned that the behavior of the defaulted accounts where very different from the other periods. KS -statistic was also slightly improved but not as significant as the other metrics.

Inhouse model UCBlanco

En bild som visar text, diagram, linje, skärmbild

Automatiskt genererad beskrivningEn bild som visar text, diagram, linje, Graf

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, linje, Graf

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, linje, Graf

Automatiskt genererad beskrivning

# 8. Selection of model continued – variables

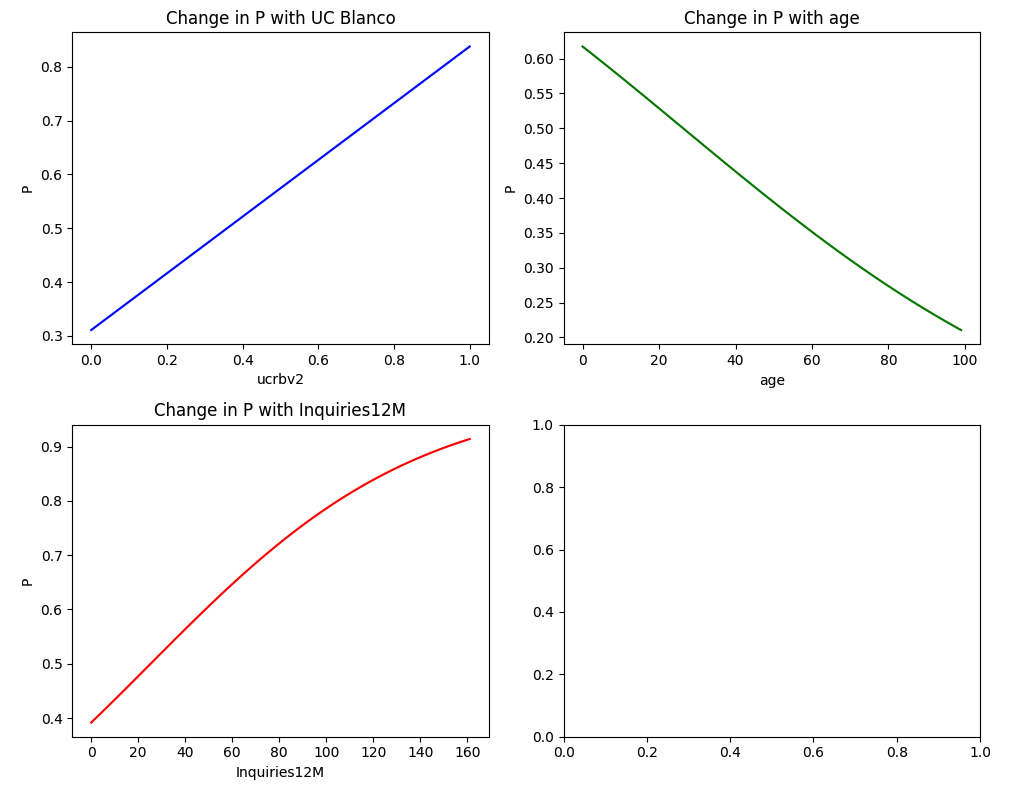
The final model has four features:

- UC Blanco Score

- Applicants age

- Number of UC-inquiries last 12 months

The table below shows the coefficients used to calculate the expected probability of default. A positive number = higher credit risk, ie a higher UC-score and more inquiries indicates a higher risk, while age and mortgage loan amount indicates a lower risk. These patterns are also shown in the graphs to the right.



|  |  |
| --- | --- |
| **Variable** | **Coefficient** |
| Intercept | -0.27028366 |
| UCB-score | 2.44117257 |
| Applicants age | -0.01818046 |
| Inquiries 12 months | 0.01741601 |

# 9. Calibration

A calibration of the pd level the model gives is necessary since the model will give a pd based on the development period, in this case defaults between 2019-2023. That period does not cover full credit cycle and contains significant volatility.

We have observed the macro development during this the period but cannot find correlation between a specific macro-variable and default levels. This is likely since our underwriting strategy has been changing over time and rather is the driver behind movements. With that said, increasing interest rates and inflation is of course a reason behind high losses last 12 months.

Without calibration – the pd-level from modelling sample would be at 5%, which we think is too low given current situation in the market. Thus, we have calibrated it up to 8%. When macro environment shifts and we observe lower levels than expected, this can be adjusted downwards.

En bild som visar text, skärmbild, Teckensnitt, linje

Automatiskt genererad beskrivning

Since the trained model underwent up sampling and improved Data Science metrics, although not exceeding 0.8 in metrics like AUROC, this leads to an expected probability of default (PD) of approximately 50 percent for the model's predictions. While the model shows enhanced DS metrics, the PD it predicts may not be directly applicable. To address this, an additional model can be introduced to predict only the output from the first model.

To align the predicted probabilities with the 12-month 90+ delinquency levels reported in recent periods, the positive class's target feature can be set to a specific threshold value instead of using a binary 1. For instance, to calibrate towards a 10% level, the target feature during model training can be set at 0.1 instead of 1. This adjustment allows for a more accurate representation of the desired prediction threshold.

*Non / Calibrated Model*

En bild som visar text, skärmbild, diagram, Graf

Automatiskt genererad beskrivning

The new score from 2019. One can see that its better calibrated (follow the actual 90+) with time than any other score applied in Nstart previously.

En bild som visar text, Graf, linje, skärmbild

Automatiskt genererad beskrivning

# Risk Grouping

To analyse the portfolio has been divided into segments where is the lowest risk and 5 is highest. Expected PD per group is as follow:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | G0 | G1 | G2 | G3 | G4 | G5 |
| Min PD | 0% | 4% | 6% | 8% | 10% | 14% |
| Max PD | 4% | 6% | 8% | 10% | 14% | 100% |

Distribution over time can be seen in the lower graph.

En bild som visar Färggrann, skärmbild, linje, aqua

Automatiskt genererad beskrivning

Calibration within the groups:

En bild som visar linje, Graf, text, diagram

Automatiskt genererad beskrivning